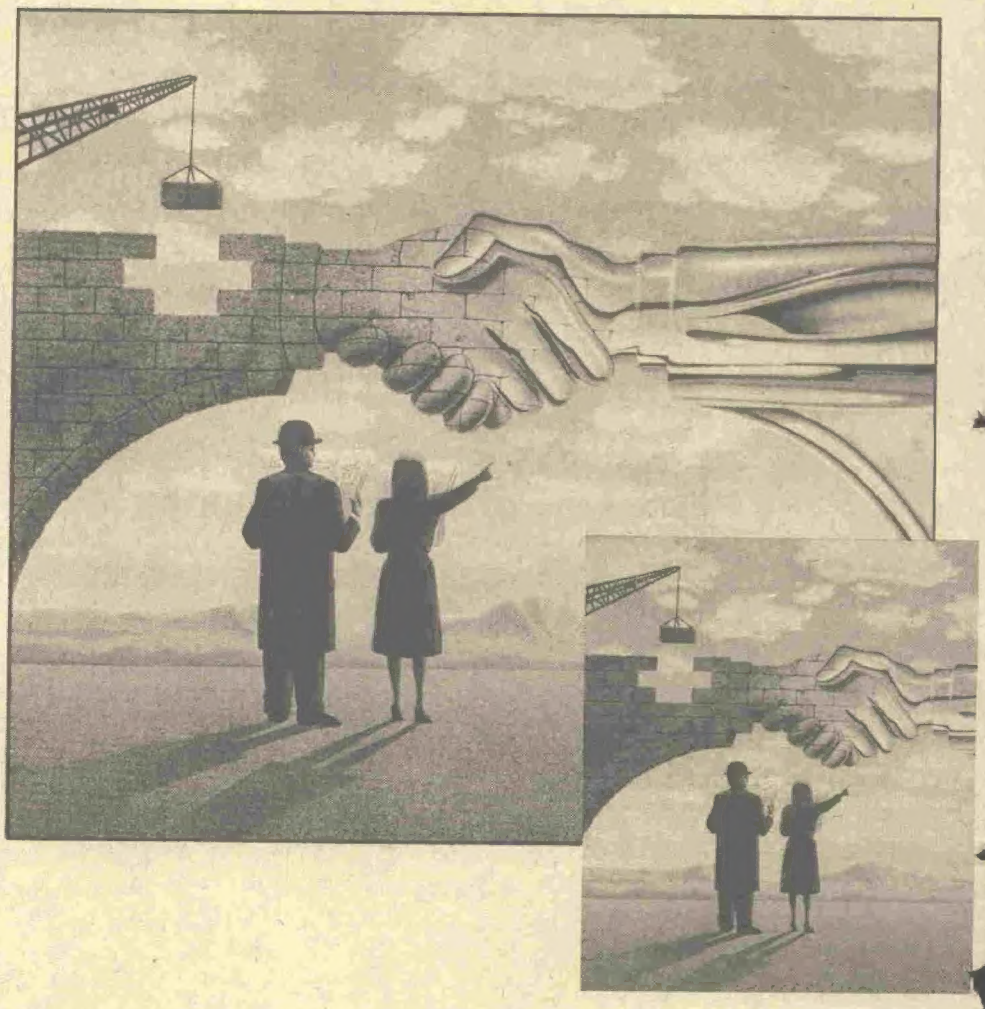
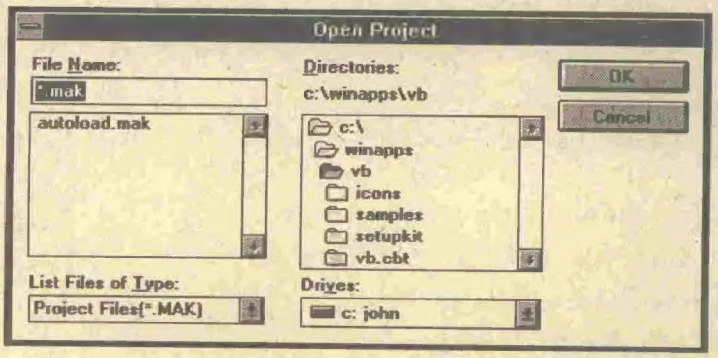
ขจร โรจนเมธินทร์

Visual Basic 2.0
:วิธีการสร้างไฟล์ ไดเรคทอรี และไดรฟ์ลิสต์บ็อกซ์

นำเสนอวิธีการสร้างไฟล์ ไดเรคทอรี และไดรฟ์ลิสต์บ็อกซ์ ซึ่งสามารถนำมาใช้ประโยชน์ได้มาก



คอนโทรลชนิดหนึ่งที่วิชวลเบสิกมีให้ใช้ ได้แก่ ลิสต์บ็อกซ์ แต่สำหรับลิสต์บ็อกซ์ที่จะกล่าวถึงในบทความนี้เป็นลิสต์บ็อกซ์ที่มีคุณสมบัติพิเศษต่างจากลิสต์บ็อกซ์ทั่วไป เนื่องจากมีหน้าที่เกี่ยวข้องกับไฟล์ ไดเรคทอรี และไดรฟ์โดยเฉพาะ ซึ่งปกติแล้วลิสต์บ็อกซ์เหล่านี้จะพบได้ในไดอะล็อกบ็อกซ์ประเภท Open, Save, Save As (ดูรูปที่ 1) การใช้งานไฟล์ลิสต์บ็อกซ์ ไดเรคทอรีลิสต์บ็อกซ์ และไดรฟ์ลิสต์บ็อกซ์ (ในบทความนี้ต่อไปจะเรียกลิสต์บ็อกซ์ธรรมดา) ในวิชวลเบสิกนี้ทำได้ง่าย เพราะว่ามี Event Procedure และ Properties ที่เราจะนำมาใช้ได้ทันที บทความตอนนี้จะกล่าวถึงวิธีสร้างแอพพลิเคชันที่นำลิสต์บ็อกซ์มาใช้งานประกอบ โดยประยุกต์เข้ากับการดิสเพลย์ภาพกราฟิก ซึ่งผู้เขียนเคยกล่าวถึงในตอนสร้างเมนูแล้ว



รูปที่ 1 แสดงลักษณะของไดอะล็อกบ็อกซ์ Open ที่ประกอบด้วยลิสต์บ็อกซ์ทั้งสามชนิด

เริ่มใช้ลิสต์บ็อกซ์

เราจะเริ่มทำความเข้าใจในการใช้ลิสต์บ็อกซ์โดยทดลองสร้างโปรแกรมตัวอย่างขึ้นมาก่อน โดยเลือกคอนโทรล

ทั้งสามชนิดจากทูลบาร์และเพิ่มเท็กซ์-บ็อกซ์เพื่อใช้เลือกเฉพาะกลุ่มไฟล์ที่ต้องการขึ้นมาแสดงในไฟล์ลิสต์บ็อกซ์ คอนโทรลทั้งหมดจะนำมาประกอบกันในฟอร์มตามรูปที่ 2 จะเห็นว่าวิชวลเบสิกจะนำค่าปัจจุบันของลิสต์บ็อกซ์แต่ละชนิดขึ้นมาแสดงทันที เช่น ในไฟล์ลิสต์บ็อกซ์ก็จะนำรายชื่อไฟล์ทั้งหมดในไดเรคทอรี C:\WINAPPS\VB ซึ่งเป็นไดเรคทอรีปัจจุบันขึ้นมาแสดง เป็นต้น ต่อไปเราจะเริ่มเขียน Event Procedure และกล่าวถึง Properties ที่เกี่ยวกับคอนโทรลทั้งสามที่สำคัญ

Event ที่เราต้องเกี่ยวข้องเมื่อใช้ลิสต์บ็อกซ์ก็คือ Change เพราะจะเป็น Event Procedure ที่ถูกเรียกมาทำงานทุกครั้งที่มีการเปลี่ยนแปลงในลิสต์บ็อกซ์ เช่น ถ้าผู้อ่านคลิ๊กเม้าส์ที่บัททอนของไดรฟ์ลิสต์บ็อกซ์ Drive1 โพรซีเดอร์ Drive1__Change( ) จะคอยรับผิดชอบต่อกิจกรรมที่ผู้อ่านทำ ซึ่งก็คือ การเลือกชื่อไดรฟ์ใหม่นั่นเอง การเลือกไดรฟ์ใหม่นี้จะทำให้ Properties ของไดรฟ์ลิสต์บ็อกซ์ คือ Drive เปลี่ยนค่าไป ดังนั้นเราจะต้องเขียนรายละเอียดของโพรซิเดอร์ดังนี้

```
Sub Drive1__Change ( )
    Dir1.Path = Drive1.Drive
End Sub
```

จากโพรซีเดอร์นี้จะเห็นว่าเราต้องกำหนดค่าพาธที่เป็น Properties ของไดเรคทอรีลิสต์บ็อกซ์ Dir1 ด้วยชื่อไดรฟ์ใหม่ เพื่อให้ข้อมูลใหม่ของ Drive1 มาแสดงใน Dir1 การกระทำดังกล่าวจะทำให้เกิดการเปลี่ยนแปลงใน Dir1 ตามมา ทำให้โพรซีเดอร์ Dir1__Change ( ) ถูกเรียกมาทำงานทันที นอกจากนี้หากผู้อ่านดับเบิ้ลคลิ๊กเม้าส์ที่ชื่อสับไดเรคทอรีที่ต้องการก็จะทำให้ Dir1__Change ( ) ทำงานเช่นกัน ซึ่งเราจะต้องเขียนโพรซีเดอร์ตัวนี้ใหม่ดังนี้

```
Sub Dir1__Change ( )
    File1.Path = Dir1.Path
End Sub
```

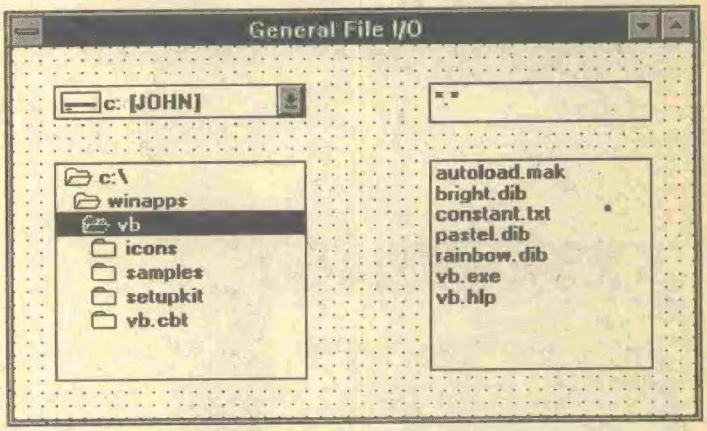
ในโพรซีเดอร์ Dir1__Change ( ) ก็เพียงแต่นำค่า Dir1.Path ที่ถูกเปลี่ยนจาก Drive1.Change ( ) หรืออาจเกิดจากผู้ใช้เลือกเองโดยตรงมากำหนดค่าให้กับพาธที่เป็น Properties ของไฟล์ลิสต์บ็อกซ์ File1 ทำให้ข้อมูลทั้งหมดของ File1 เปลี่ยนแปลง โดยนำเอารายชื่อไฟล์ทั้งหมดที่อยู่ในไดเรคทอรีใหม่มาเก็บไว้ในลิสต์บ็อกซ์แทนนั่นเอง

การเลือกชื่อไฟล์ที่ต้องการจากไฟล์ลิสต์บ็อกซ์ File1 ซึ่งปกติก็จะต้องดับเบิ้ลคลิ๊กเม้าส์ที่ชื่อไฟล์ต้องการ ซึ่งเราก็ต้องเขียน Event Procedure เพื่อรับผิดชอบกิจกรรมนี้ คือ File1__DblClick ( ) ดังนี้

```
Sub File1__DblClick ( )
    MyPath = Dir1.Path
    If Right(MyPath, 1) <> "\" Then
        MyPath = MyPath + "\"
    End If
    FullFileName = MyPath + File1.FileName
    MsgBox FullFileName
End Sub
```

ใน File1__DblClick ( ) จะตรวจสอบว่ามีเครื่องหมาย "\" ที่ชื่อพาธหรือไม่ โดยใช้ฟังก์ชัน Right ( ) ตรวจสอบสตริง MyPath ถ้าไม่มีก็จะเพิ่มเครื่องหมาย "\" ให้ จากนั้นก็นำชื่อพาธมาเชื่อมกับชื่อไฟล์ใน File1.FileName ให้เป็นชื่อเต็ม เก็บไว้ในตัวแปร FullFileName แล้วจึงนำชื่อไฟล์ที่เลือกมาแสดงในเมสเสจบ็อกซ์ ดังรูปที่ 3

สำหรับการกำหนดกลุ่มไฟล์เฉพาะโดยใช้ Wildcard ซึ่งในโปรแกรมตัว-



รูปที่ 2
แสดงส่วนประกอบของฟอร์มตัวอย่างที่มีลิสต์บ็อกซ์ทั้งสามชนิด

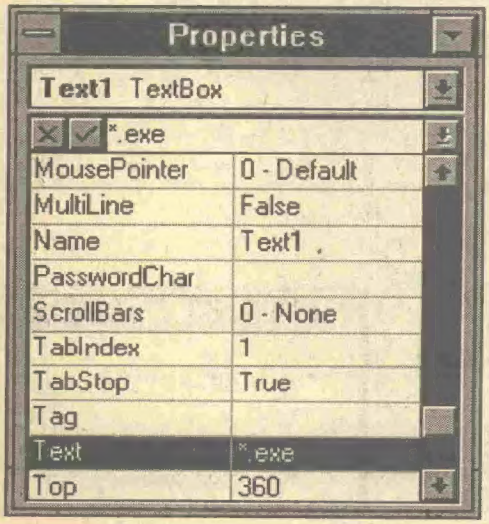
รูปที่ 3
แสดงเมสเสจบ็อกซ์ที่รายงานชื่อไฟล์ที่ผู้ใช้เลือกจากไฟล์ลิสต์บ็อกซ์

อย่างนี้เราจะทำในเท็กซ์บ็อกซ์ Text1 ที่เรากำหนดค่าเริ่มต้นไว้เป็น *.exe ใน Properties Text ของ Text1 ตอนที่สร้างคอนโทรลเลย (ดูรูปที่ 4) หากเรากำหนดแค่ค่าเริ่มต้นของ Text1.Text เท่านั้น โปรแกรมจะยังทำงานไม่ถูกต้อง เราต้องเขียนโพรซีเดอร์ Form_Load ( ) ใหม่ และให้แก้ไข Properties Pattern ของ File1 ตอนโปรแกรมเริ่มทำงานดังนี้

```
Sub Form_Load ( )
    File1.Pattern = Text1.Text
End Sub
```

และโพรซีเดอร์ที่เราต้องแก้ไขอีกอันหนึ่งก็คือ Text1_Change( ) ที่เป็นโพรซีเดอร์ที่จะถูกเรียกมาทำงานเมื่อมีการแก้ไขค่าเท็กซ์ในเท็กซ์บ็อกซ์ Text1 เช่นเดียวกับที่เขียนใน Form_Load ดังนี้

```
Sub Text_Change( )
    File1.Pattern = Text1.Text
End Sub
```



รูปที่ 4
แสดงหน้าต่าง Properties ของคอนโทรลเท็กซ์บ็อกซ์เมื่อขณะแก้ไขค่าเริ่มต้นของ Properties Text เป็น *.EXE

หน้าที่ของแอพพลิเคชัน

Icon Viewer

ในแอพพลิเคชันนี้เราจะใช้ส่วนประกอบต่าง ๆ ของโปรแกรมตัวอย่างที่กล่าวถึงข้างต้น ได้แก่ ไฟล์ลิสต์บ็อกซ์ ไดเรคทอรีลิสต์บ็อกซ์ และไดรฟ์ลิสต์บ็อกซ์ นำมาประกอบกันเพื่อจะใช้เลือกชื่อไฟล์ที่ต้องการ โดยไฟล์ดังกล่าวจะเป็นชื่อไอคอนซึ่งมีนามสกุล ICO ซึ่งปกติจะมีมาให้ในไดเรคทอรี \VB\ICONS ของวิชวลเบสิกแล้ว ดังนั้นเราจะกำหนดค่าเริ่มต้นของ File1.Pattern ให้เป็น *.ICO และเมื่อผู้ใช้เลือกไฟล์ที่ต้องการจากไฟล์ลิสต์บ็อกซ์แล้ว เราก็จะนำไอคอนที่เก็บอยู่ในไฟล์นั้นมาดิสเพลย์ในกรอบ Picture1 อีกที

เริ่มสร้างโปรแกรม

Icon Viewer

เริ่มต้นก็ต้องสร้างฟอร์มของโปรแกรมก่อน ฟอร์มของ Icon Viewer นี้จะต่างจากตัวอย่างก่อน (ดูรูปที่ 5) โดยมีคอนโทรลเพิ่ม คือ เพิ่มพิกเจอร์บ็อกซ์ Picture1 ไว้ใช้ดิสเพลย์ไอคอนที่ผู้ใช้เลือก เลเบลอีก 5 อัน และคอมมานด์บัททอน 1 ปุ่ม

ค่าเริ่มต้นของ Properties ของฟอร์มที่ต้องกำหนดใหม่ก็คือ Caption ที่จะต้องเปลี่ยนเป็น "Icon Viewer", MaxButton จะกำหนดให้เป็น False เพื่อให้ไม่มีปุ่มสำหรับขยายหน้าต่างของโปรแกรมให้ใหญ่สุด ซึ่งเราจะเห็นเมื่อโปรแกรมทำงานแล้วเท่านั้น, BorderStyle จะกำหนดเป็นแบบ 1-Fixed Single คือเราจะบังคับให้หน้าต่างของโปรแกรมไม่มีขอบซึ่งเราจะใช้แก้ไขขนาดหน้าต่าง

ต่อไปเราจะแก้ไขเฉพาะ Properties Caption ของเลเบลทั้ง 5 อัน ดังนี้ Label1 เป็น c:\winapps\vb\icons ซึ่งเป็นชื่อไดเรคทอรีที่ใช้เก็บไฟล์ไอคอนทั้งหมดของวิชวลเบสิกบนเครื่องของผู้เขียน โดย Caption นี้จะเปลี่ยนแปลงค่าไปตามชื่อไดเรคทอรีที่ผู้อ่านเลือกเมื่อโปรแกรมทำงาน, Label2 เป็น Directories:, Lebel3 เป็น Filenames:, Label4 เป็น *.ico ซึ่งค่า Caption นี้จะเปลี่ยนตามชื่อไฟล์ที่ผู้อ่านเลือกเช่นกัน, Label5 เป็น Drives: และสุดท้ายเป็น Caption ของคอมมานด์บัททอน Command1 เป็น Exit

หลังจากที่กำหนดค่าเริ่มต้นของ Properties ต่าง ๆ เรียบร้อยแล้ว ก็จะเริ่มต้นเขียน Event Procedure ต่าง ๆ ต่อ ได้แก่ Drive1_Change ( ) โดยเขียนให้มีรายละเอียดเหมือนโปรแกรมในตัวอย่างก่อน Dir1_Change ( ) นี้จะเพิ่มการกำหนด Caption ของ Label1 ให้มีค่าตามไดเรคทอรีที่ผู้ใช้เลือกด้วยค่า Dir1.Path ดังนี้

```
Sub Dir1_Change ( )
    File1.Path = Dir1.Path
    Label1.Caption = Dir1.Path
End Sub
```

ดังนั้นเมื่อผู้ใช้เลือกไดเรคทอรีใหม่ ชื่อไดเรคทอรีก็จะถูกนำมาแสดงบนหน้าต่างที่ตำแหน่งใต้ Directories: แต่จะมีปัญหาตรงที่ถ้าชื่อไดเรคทอรียาวเกินไปก็ไม่สามารถนำมาแสดงได้หมด เพราะจะเกินขนาดหน้าต่างที่เรากำหนดไว้ โพรซีเดอร์ต่อไปที่จะต้องเขียนก็คือ File1_Click ( ) ก็ยังคล้ายของเดิม แต่ในโพรซีเดอร์นี้จะนำชื่อไฟล์ที่ผู้ใช้เลือกส่งให้ฟังก์ชัน LoadPicture( ) เพื่อจะดิสเพลย์ไอคอนในคอนโทรลพิกเจอร์ดังนี้

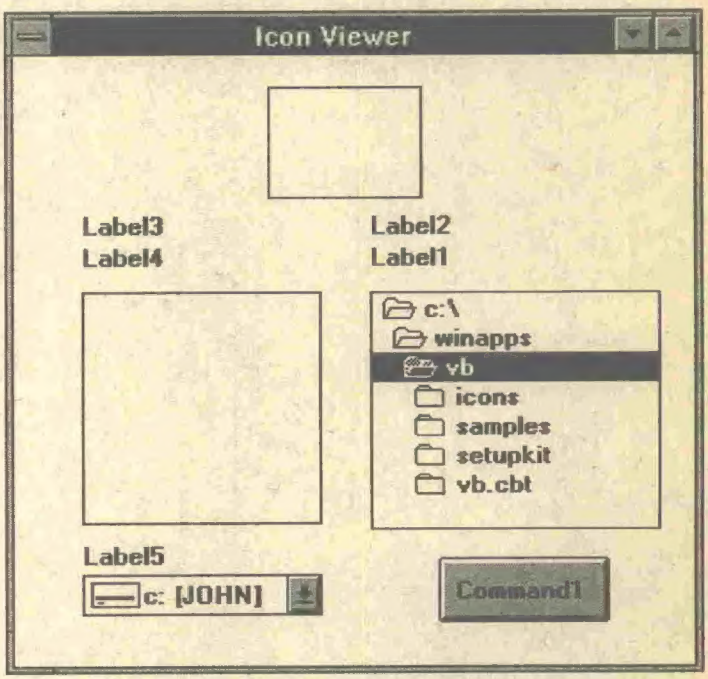
```
Sub File1_Click ( )
    Label4.Caption = File1.FileName
    MyPath$ = Dir1.Path
    If Right(MyPath$, 1) <> "\" Then
      MyPath$ = MyPath$ + "\"
    End If
    IconFile$ = MyPath$ + File1.FileName
    Picture1.Picture = LoadPicture(IconFile$)
End Sub
```

เมื่อผู้ใช้คลิ๊กเม้าส์ที่ชื่อไฟล์ไอคอนที่ต้องการ ช่องพิกเจอร์ในหน้าต่างก็จะมีรูปไอคอนแสดง หรือหากกดคีย์ลูกศรเพื่อเลื่อนแถบสว่างขึ้นลงไปที่ไฟล์ที่ต้องการ ภาพของไอคอนก็จะเปลี่ยนไปตามชื่อไฟล์ที่ผู้ใช้เลือกด้วย นอกจากนี้โพรซีเดอร์ที่ต้องเขียนอีกสองอันก็คือ Form_Load ( ) และ Command1_Click ( )

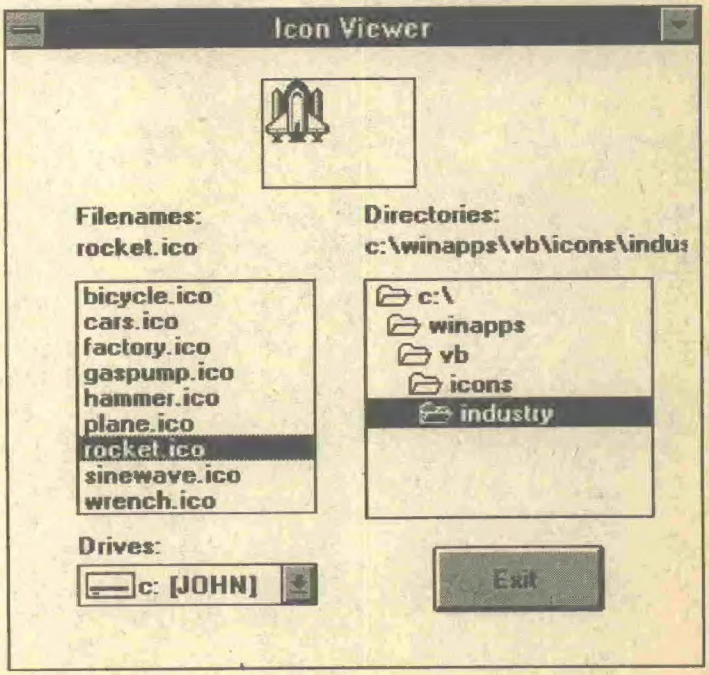
```
Sub Form_Load ( )
    Label1.Caption = Dir1.Path
End Sub
```

```
End Sub
Sub Command1_Click ( )
    End
End Sub
```

หน้าที่ของโพรซีเดอร์ Form_Load ( ) ก็คือ นำชื่อไดเรคทอรีปัจจุบันขณะที่โปรแกรมเริ่มทำงานมาแสดงที่ตำแหน่งใต้ Directories: สำหรับ Command1_Click ( ) นั้นจะใช้จบการ



รูปที่ 5 แสดงคอนโทรลต่าง ๆ ของฟอร์ม Icon Viewer



รูปที่ 6 แสดงลักษณะหน้าต่างเมื่อรันโปรแกรม Icon Viewer

```
VERSION 2.00
Begin Form Form1
  BackColor       = &H00FFFFFF&
  BorderStyle     = 1 'Fixed Single
  Caption         = "Icon Viewer"
  ForeColor       = &H00E0FFFF&
  Height          = 5070
  Icon            = ICV.FRX:0000
  Left            = 1035
  LinkTopic       = "Form1"
  MaxButton       = 0 'False
  ScaleHeight     = 4665
  ScaleWidth      = 4815
  Top             = 1140
  Width           = 4935
  Begin CommandButton Command1
    BackColor     = &H00FFFFFF&
    Caption       = "Exit"
    Height        = 495
    Left          = 3000
    TabIndex      = 5
    Top           = 3840
    Width         = 1215
  End
  Begin DriveListBox Drive1
    Height        = 315
    Left          = 480
    TabIndex      = 3
    Top           = 3960
    Width         = 1695
  End
  Begin DirListBox Dir1
    Height        = 1830
    Left          = 2520
    TabIndex      = 2
    Top           = 1800
    Width         = 2055
  End
  Begin FileListBox File1
    Height        = 1785
    Left          = 480
    Pattern       = "*.ico"
    TabIndex      = 1
    Top           = 1800
    Width         = 1695
  End
  Begin PictureBox Picture1
    Height        = 855
    Left          = 1800
    ScaleHeight   = 825
    ScaleWidth    = 1065
    TabIndex      = 0
    Top           = 240
    Width         = 1095
  End
  Begin Label Label5
    Caption       = "Drives:"
    Height        = 255
    Left          = 480
    TabIndex      = 9
    Top           = 3720
    Width         = 1575
  End
  Begin Label Label4
    Caption       = "*.ico"
    Height        = 255
    Left          = 480
    TabIndex      = 8
    Top           = 1440
    Width         = 1695
  End
  Begin Label Label3
    Caption       = "Filenames:"
    Height        = 255
    Left          = 480
    TabIndex      = 7
    Top           = 1200
    Width         = 1695
  End
  Begin Label Label2
    Caption       = "Directories:
    Height        = 255
    Left          = 2520
    TabIndex      = 6
    Top           = 1200
    Width         = 1455
  End
  Begin Label Label1
    Caption       = "c:\winapps\vb\icons"
    Height        = 255
    Left          = 2520
    TabIndex      = 4
    Top           = 1440
    Width         = 2295
  End
End

Sub Command1_Click ()
  End
End Sub

Sub Dir1_Change ()
  File1.Path = Dir1.Path
  Label1.Caption = Dir1.Path
End Sub

Sub Drive1_Change ()
  Dir1.Path = Drive1.Drive
End Sub

Sub File1_Click ()
  MyPath$ = Dir1.Path
  Label4.Caption = File1.FileName
  If Right(MyPath$, 1) <> "\" Then
    MyPath$ = MyPath$ + "\"
  End If
  IconFile$ = MyPath$ + File1.FileName
  Picture1.Picture = LoadPicture(IconFile$)
End Sub

Sub Form_Load ()
  Label1.Caption = Dir1.Path
End Sub
```

รูปที่ 7
แสดงโค้ดของโปรแกรม Icon Viewer ทั้งหมด

ทำงานของโปรแกรมเมื่อผู้ใช้คลิ๊กเม้าส์ที่ปุ่ม Exit นั่นเอง เมื่อถึงจุดนี้โปรแกรม Icon Viewer ก็พร้อมที่จะทำงานแล้ว ขอให้ผู้อ่านลองกดคีย์ F5 เพื่อรันโปรแกรมดู ก็จะได้ผลตามรูปที่ 6

จบบทเรียน

ในบทความนี้ผู้เขียนได้นำเสนอวิธีการสร้างไฟล์ ไดเรคทอรี และไดรฟ์ลิสต์บ็อกซ์ ซึ่งเป็นสิ่งที่ผู้อ่านสามารถนำมาใช้ประโยชน์ได้มาก เนื่องจากลิสต์บ็อกซ์ทั้งสามชนิดมักต้องมีประกอบในโปรแกรมทุกโปรแกรม ผู้อ่านคงจะเห็นด้วยว่าคอนโทรลทั้งสามที่วิชวลเบสิกมีมาให้ใช้นี้สร้างและเขียนโพรซีเดอร์ได้ง่ายมาก หวังว่าคงจะเป็นประโยชน์กับผู้อ่านพอสมควร